ISO/CD TS 18876-1

ISO TC184/SC4/WG10 N337 **Date: 2001-06-01**

Supersedes: ISO TC184/SC4/WG10 N333

Industrial automation systems and integration — Integration of industrial data for exchange, access, and sharing — Part 1: Architecture overview and description

ABSTRACT:

This document provides an overview and description of the ISO 18876 architecture for integration of industrial data.

KEYWORDS:

industrial data, integration, exchange, access, sharing, architecture, overview

COMMENTS TO READER:

This is the committee draft of Part 2 of ISO 18876. This document has been reviewed using the internal review checklist (see WG10 N339), the project leader checklist (see <provide N-number>) and the convener checklist (see <provide N-number>), and has been determined to be ready for this ballot cycle.

Reviewers should submit comments on this document via their national standards bodies..

Project leader:	Matthew West Operations & Asset Management Shell Services International H3229, Shell Centre London, SE1 7NA UK	Part editor:	Matthew West Operations & Asset Management Shell Services International H3229, Shell Centre London, SE1 7NA UK
Telephone:	+44 207 934 4490	Telephone:	+44 207 934 4490
Fax:	+44 207 934 7929	Fax:	+44 207 934 7929
Email:	Matthew.R.West@is.shell.com	Email:	Matthew.R.West@is.shell.com

Document type:	Technical Specification	Document subtype:	Not applicable
Document stage:	Committee Draft (30)	Document language:	E
File name:	wg10n337.doc	Template:	ISO SC4 latest.dot

Contents

Figures

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 3

Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75% of the member bodies casting a vote.

In other circumstances, particularly when there is an urgent market requirement for such documents, a technical committee may decide to publish other types of normative document:

— an ISO Publicly Available Specification (ISO/PAS) represents an agreement between technical experts in an ISO working group and is accepted for publication if it is approved by more than 50 % of the members of the parent committee casting a vote;

— an ISO Technical Specification (ISO/TS) represents an agreement between the members of a technical committee and is accepted for publication if it is approved by 2/3 of the members of the committee casting a vote.

An ISO/PAS or ISO/TS is reviewed every three years with a view to deciding whether it can be transformed into an International Standard.

Attention is drawn to the possibility that some of the elements of this part of ISO 18876 may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights.

ISO/TS 18876-1 was prepared by Technical Committee ISO/TC184, *Industrial automation systems and integration*, Subcommittee SC4, *Industrial data*.

This International Standard is organized as a series of parts, each published separately. The structure of this International Standard is described in this part of ISO 18876.

A complete list of parts of ISO 18876 is available from the Internet:

`<http://www.iso18876.org/parts.html>`

Annex A forms a normative part of this part of ISO 18876.

0 Introduction

0.1 Overview of ISO 18876

This International Standard establishes an architecture, a methodology, and other specifications for integrating industrial data for exchange, access, and sharing. The following activities are supported:

- integrating data which may be:
  - from different sources or different contexts,
  - described by different models, or
  - defined in different modelling languages;
- sharing data among applications through systems integration architectures;
- resolving conflict between models developed with different objectives;
- translating data between different encodings;
- translating models between different modelling languages.

The components that support these activities include:

- integration models;
- methods for creating, extending, and updating integration models;
- methods for creating a mapping specification to map data instances between an integration model and an application model that falls within its scope;
- encoding and decoding of data and models with different formats, such as SGML [1], XML [7], EXPRESS [2], UML [5] and ISO 10303-21 [4];
- methods for consolidating data sets from different sources and different models ;
- appropriate modelling and mapping languages.

0.2 Organization of this part of ISO 18876

This part of ISO 18876 is organized as follows:

- clause 1 specifies the scope and field of application of the International Standard and of this part of ISO 18876;
- clause 2 identifies additional standards that, through references in this part of ISO 18876, constitute provisions of this part of ISO 18876;
- clause 3 defines terms and abbreviations used in this part of ISO 18876;
- clause 4 describes the organization of this International Standard;
- clause 5 describes the fundamental concepts and assumptions on which this International Standard is based;
- clause 6 provides an overview of the model integration process;

— clause 7 identifies some components of the integration architecture;

— clause 8 provides an overview of the processes of data mapping and consolidation;

— clause 9 summarizes the relationships with other standards.

0.3 Target audiences

The target audiences for this part of ISO 18876 are as follows:

— technical managers wishing to determine whether ISO 18876 is appropriate for their business needs;

— implementers wishing to obtain an overview of its contents.

Industrial automation systems and integration — Integration of industrial data for exchange, access, and sharing — Part 1: Architecture overview and description

1 Scope

This International Standard establishes an architecture, a methodology, and other specifications for integrating industrial data for exchange, access, and sharing. The following activities are supported:

- integrating data which may be:
  - from different sources or different contexts,
  - described by different models, or
  - defined in different modelling languages;
- sharing data among applications through systems integration architectures;
- resolving conflict between models developed with different objectives;
- translating data between different encodings;
- translating models between different modelling languages.

The following are within the scope of ISO 18876:

- integration models;
- methods for creating, extending, and updating integration models;
- methods for creating a mapping specification to map data instances between an integration model and an application model that falls within its scope;
- encoding and decoding of data and models with different formats;
- methods for consolidating data sets from different sources and different models ;
- appropriate modelling and mapping languages.

The following is within the scope of this part of ISO 18876:

- an outline of the architecture.

The following are outside the scope of this part of ISO 18876:

- detailed specifications of the elements of the architecture.

NOTE Such specifications can be found in other parts of ISO 18876 or in other standards.

2 Normative references

The following normative documents contain provisions which, through reference in this text, constitute provisions of this part of ISO 18876. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this International Standard are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references, the latest edition of the normative document referred to applies. Members of ISO and IEC maintain registers of currently valid International Standards.

ISO/IEC 8824-1:1994, *Information technology — Open systems interconnection — Abstract syntax notation one (ASN.1) — Part 1: Specification of basic notation*.

ISO 10303-1:—[1)], *Industrial automation systems and integration — Product data representation and exchange — Part 1: Overview and fundamental principles.*

3 Terms, definitions, and abbreviations

3.1 Terms and definitions

For the purposes of this part of ISO 18876, the following terms, definitions, and abbreviations apply; those taken from ISO 10303-1 are repeated below for convenience.

NOTE 1 Definitions copied verbatim from other standards are followed by a reference to the standard in brackets, such as "[ISO 10303-1]". In these cases the definition in the referenced document is normative; its repetition here is informative and in the case of any discrepancy the definition in the referenced document has precedence. An explanatory note follows definitions that have been adapted from other standards. In these cases, the definition given here is normative for the purposes of this part of ISO 18876.

NOTE 2 A glossary of terms and definitions used in this International Standard is available on the Internet:

`<http://www.iso18876.org/glossary.html>`

3.1.1
application model (AM)
model that represents information used for some particular purpose

NOTE Some application models are also integration models (see 3.1.12).

3.1.2
class
collection to which some significance is attached

EXAMPLE Pump, power station, engineer, and fictional space vehicle are examples of classes.

3.1.3
concept
general notion or idea of something

[1)] To be published. (Revision of ISO 10303-1:1994)

3.1.4
data
representation of information in a formal manner suitable for communication, interpretation, or processing by human beings or computers

[ISO 10303-1]

3.1.5
data model
definition, structure, and format of data

3.1.6
derived concept
concept in an integration model that is wholly defined in terms of primitive concepts

3.1.7
encoding transformation
transformation of the way data elements are represented for computer processing

EXAMPLE Conversion of data governed by an EXPRESS schema from an ISO 10303-21 file to an XML document is an example of an encoding transformation.

3.1.8
foundation concept
primitive concept that determines the underlying world viewpoint of an integration model

EXAMPLE The concepts of class and individual are foundation concepts for a general integration model.

3.1.9
general concept
primitive concept that has very wide applicability, but is a specialization of some foundation concept

NOTE The boundary between a foundation concept and a general concept may be arbitrary; some concepts may be thought of as both foundation concepts and general concepts.

3.1.10
individual
thing that exists in space and time

NOTE This includes things that actually exist, or have existed, and things that possibly exist (past, present, and future) in space and time.

EXAMPLE The pump with serial number ABC123, Battersea Power Station, Sir Joseph Whitworth, and the Starship "Enterprise" are examples of individuals.

3.1.11
information
facts, concepts, or instructions

[ISO 10303-1]

3.1.12
integration model (IM)
application model that can represent the information that is represented by two or more application models

3.1.13
mapping specification
definition of the transformations necessary to take information according to one data model and represent the same information according to another data model

NOTE 1 A mapping specification can include data structure transformations, data value transformations, data encoding transformations, and terminology transformations.

NOTE 2 Mapping specifications can be procedural, or declarative, or a combination of these.

3.1.14
model
limited representation of something suitable for some purpose

3.1.15
model context
sum of constraints that limit the possible extension of a model without changing any existing declarations

NOTE This term is more general than application context as defined in ISO 10303-1.

3.1.16
model scope
range of information that an application model can describe

3.1.17
primitive concept
concept in an integration model that is not wholly defined in terms of other concepts

3.1.18
specific concept
primitive concept that is a specialization of some general concept and has a limited range of applicability

EXAMPLE Car, process plant, quark, purchase order, and XML document are examples of specific concepts.

NOTE The boundary between a general concept and a specific concept may be arbitrary; some concepts may be thought of as both general concepts and specific concepts.

3.1.19
structural transformation
transformation of the structure of data

NOTE The change in structure could be to the rearranging of attributes, the splitting of attributes across entity data types, or the creation of new attributes.

3.1.20
terminology transformation
transformation of the term used to refer to a thing

NOTE This could be between synonyms in one language, or between different languages.

3.1.21
transformation
change of form

3.1.22
view
constrained representation of a data model

3.2 Abbreviations

For the purposes of this part of ISO 18876, the following abbreviations apply:

AM application model

IM integration model

4 Organization of ISO 18876

ISO 18876 is divided into a number of parts.

ISO 18876-1, this part, provides an overview and specifies an architecture for the integration of industrial data.

ISO 18876-2 specifies methods for integrating application models and for developing and extending integration models.

NOTE Other specifications may be developed to extend the capability of ISO 18876, such as:

— models designed to integrate two or more other models;

— models designed to meet the needs of a particular application, application models;

— mapping specifications designed to specify how a data population of one model may be migrated to another model;

— mapping specifications designed to specify how a model in one language may be migrated to another language;

— methods and languages to support the definition of models and mappings between different modelling languages;

— methods and specifications for the encoding of models and transformation between encodings;

— the specification of services and interfaces to be provided by conforming implementations.

5 Fundamental concepts and assumptions

The following fundamental concepts and assumptions apply to this standard.

5.1 Integration models

5.1.1 Principles

The three-schema architecture for data models shows that for any data model it is possible to construct views on the original model.

NOTE 1 The three schema architecture is described in ISO/TR 9007 [2].

In this International Standard this principle is extended to cover other types of model and modelling languages. In the integration of models this process is reversed: a model is created for which the initial models are views. A model created in this way is an integration model with respect to the initial models in that it is capable of representing information with the scope of either or both of the original models. This is illustrated in Figure 1 below.

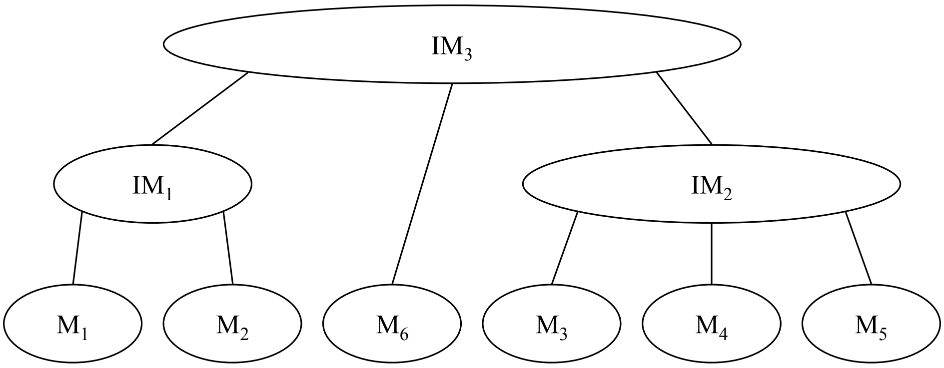

Figure 1 — Model Integration

An integration model can be created if a common understanding of the application models to be integrated can be established

NOTE 2 See Barwise & Seligman [5] for discussion of the need for common understanding.

NOTE 3 Difficulties in creating such a model point to a gap in human knowledge about the subject of the application models.

There may be more than one integration model to which an application model can be integrated, where the integration models support different ways of looking at the world. See Figure 2 below.

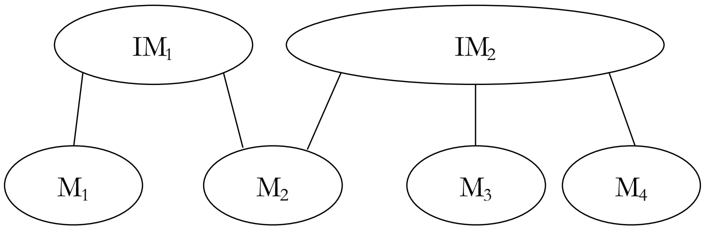

Figure 2 — Integration into more than one integration model

Models that have been created as integration models can themselves be integrated. This means that any arbitrary set of models can, in principle, be integrated at the cost of creating a new model; this is supported by the architecture defined in this standard.

NOTE 4 One possible use for the architecture defined here is the development of an integration model that is stable in the face of the integration of additional models. Here stable means that the existing integration model does not need to be changed as more models are integrated, though extensions of the integration model may be necessary.

Integration models sometimes represent concepts that are more generic than the models they integrate. This is necessarily the case when the models being integrated have conflicting constraints affecting the information that is to be represented. These constraints should be preserved by the integration process, and held in some form other than in the structure of the integration model.

NOTE 5 The constraints can be held in the mapping specification or as data within the integration model.

5.1.2 Scope and context

The scope of a model is what it actually covers. The context of a model is the sum of constraints that limit the ability to extend the scope of the model without changing any of existing declarations. These constraints include:

— activities that produce or use the data;

— organizations that produce or use the data;

— concepts that the model is “about” but are not explicitly represented in the model;

— constraints inherent in the structure of the model.

Integration models can be created that consider two or more application models of interest – the scope and context of such an integration model can be no smaller than the combined scope and context of the application models being integrated. The relationship between such an integration model and the other application models that it integrates is shown in Figure 3 below.

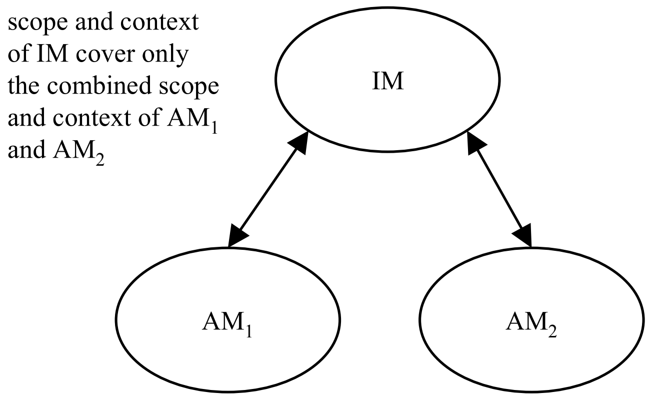

Figure 3 — A limited integration model

If requirements subsequently emerge to integrate additional application models with AM_1 and AM_2 (and hence with IM) it is unlikely that the context of these further models will fit within that of IM. This implies that IM cannot support the information represented by the further application models, and will itself have to be integrated with another integration model (created for this purpose or selected from candidate integration models). This is shown in Figure 4 below.

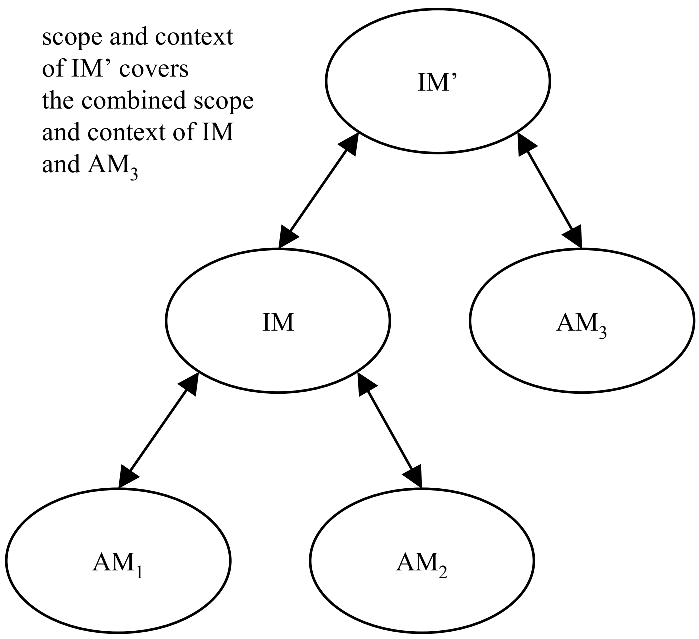

Figure 4 — Integrating an application model and a limited integration model

However, the initial integration model (IM) can be chosen to have a wide model context. This means that it can support the information needs of many different applications, even though its initial model scope is limited to that of the models that it integrates, as shown in Figure 5 below.

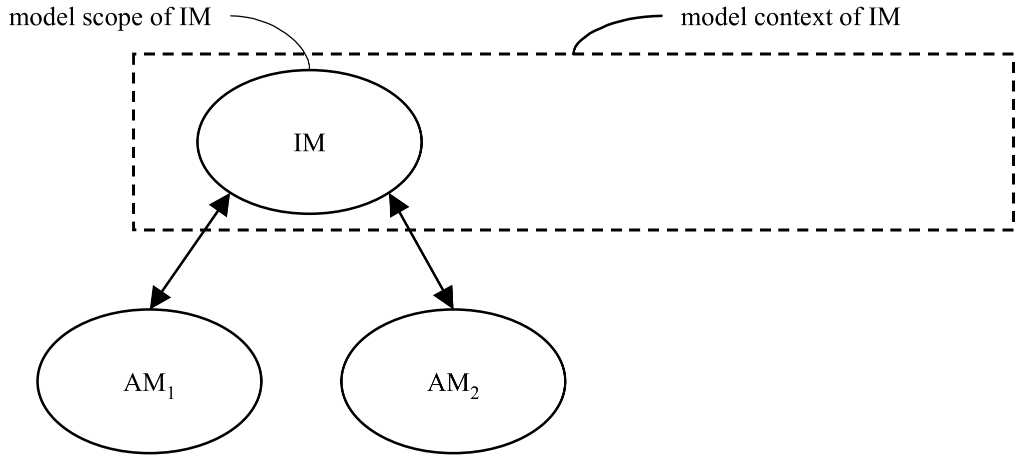

Figure 5 — Using an integration model with a wide model context

Integration of further application models can then be achieved through extension of the integration model – enlarging the model scope within the wide model context, as shown in Figure 6 below. In this case, the integration model can be extended (though the addition of constructs that represent specific and derived concepts) without changing its initial content or the mappings of the initial application models AM_1 and AM_2.

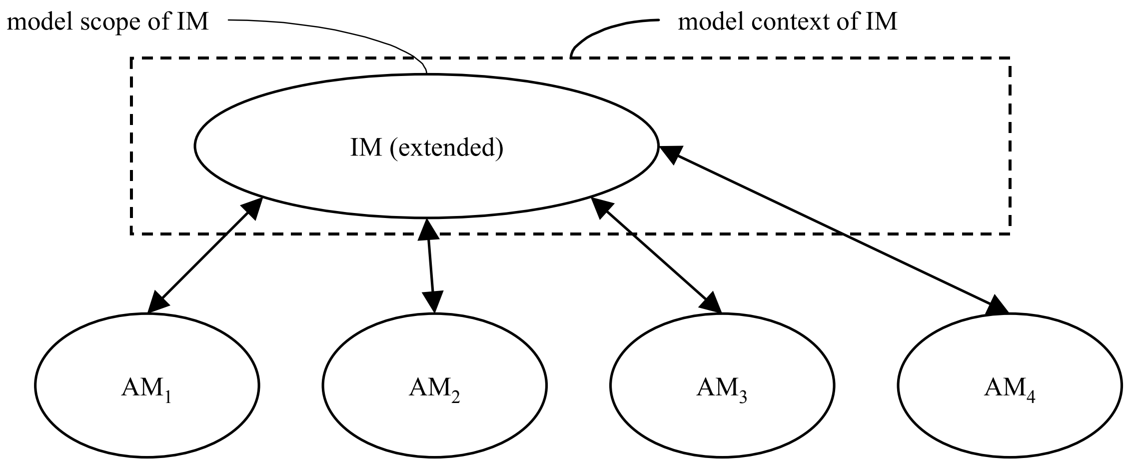

Figure 6 — Integrating additional application models

When an integration model is changed to accommodate a new application model, if the previous version of the integration model is not a subset of the new version, the previous mappings to the integration model may no longer be valid. One model is a subset of another model when each declaration specified in the first model is also a declaration in the second model. The second model is then an extension to the first.

When the changes required to the integration model are not an extension, then a new integration model should be created together with a mapping to the original integration model to preserve any mappings that may have been made to it.

A consequence of this is that it is desirable that integration models should be constructed such that integrating further application models results is an extension to the model. This is true for integration models that have a broad context. These have the following characteristics.

— Each object is modelled in terms of its underlying nature, based on a specific paradigm or view of the world being modelled.

— The names chosen for the model constructs reflect the underlying nature of the subject of the model.

— Definitions of model constructs are clear and unambiguous.

— Only those constraints that are applicable for the whole context and scope of the model are represented in the structure of the model.

— Classes are represented as nodes in a subclass/superclass (specialization/generalization) hierarchy that is rooted in a single class.

— The model represents and manages history and change.

— The modelling language chosen to represent the integration model is used consistently.

5.1.3 Integration model concepts

The concepts represented by an integration model can be classified as primitive concepts (see 3.1.17), and derived concepts (see 3.1.6). Primitive concepts are the building blocks for the definition of other concepts, and can be further classified as follows:

— foundation concepts (see 3.1.8);

— general concepts (see 3.1.9);

— specific concepts (see 3.1.18).

This is illustrated in Figure 7 below.

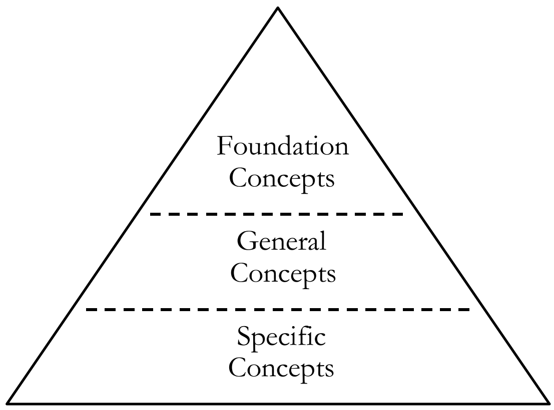

Figure 7 — Primitive Concepts

NOTE Specific concepts are dependent on general concepts that are dependent on foundation concepts, since all the lower concepts rely on the existence of one or more higher level concepts. For example, without the foundation concept of classification, there is relatively little that can be said about anything.

EXAMPLE At the top level, an integration model might have foundation concepts like classification, connection and composition. General concepts might include those of physics, and finally discipline specific concepts that are limited in their range of application.

5.1.4 A full integration model

A full integration model, as illustrated in Figure 8, is more than just primitive concepts; it includes derived concepts – useful and valid combinations of primitive concepts. Only derived concepts that are of interest need be recorded.

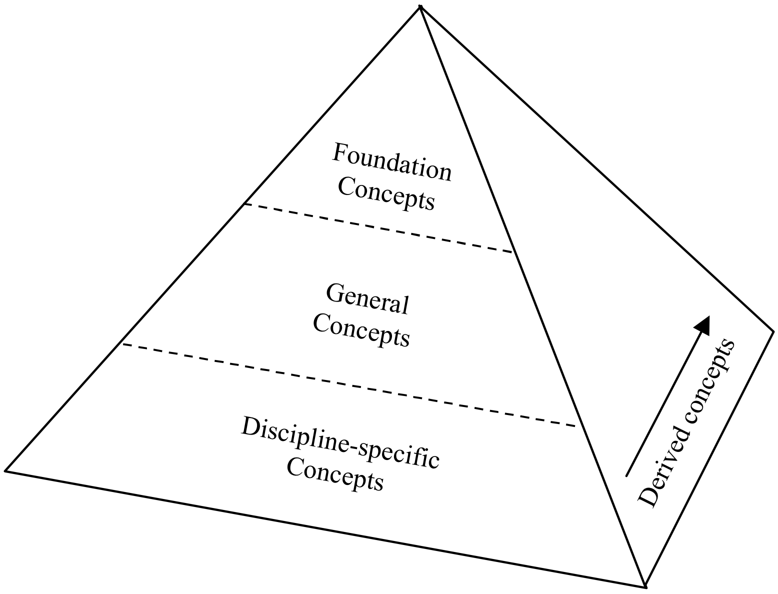

Figure 8 — A full integration model

This architecture does not require that primitive concepts are primitive forever. If a concept that is initially thought to be a primitive concept turns out not to be, then the concepts it is derived from can be identified/added, and the derivation added, so that it becomes a derived concept away from the front face of the pyramid. This allows flexibility to reflect an improved knowledge of the world, rather than reflecting knowledge of the world that is constrained by a modeller's knowledge at a point in time. Therefore, an integration model will need to be maintained and extended, and a mechanism for maintenance and extension will be necessary.

5.2 Mapping specifications

Mapping specifications specify the transformations that determine how the instances of one model can be represented as instances of another model. Mapping specifications are used in two ways, as follows.

The mapping specification can describe the mapping transformations between a subset of an integration model and a pre-existing application model that governs data that is separate from that governed by the integration model. In this case the mapping specification describes the transformations that enable assertions of equivalence of instances of one model to be made with respect to instances of the other.

The mapping specification can describe the mapping transformations between a subset of an integration model and an application model that is used as an application view. In this case the mapping specification describes how instances in the application view are created from instances in the integration model.

The following fundamental concepts apply to the mapping specifications that are created during the integration process.

— New primitive concepts are not introduced in the mapping specification; mapping specifications are limited to specifying Application Model concepts in terms of Integration Model concepts, transformations of structure, terminology, and encoding.

— Mapping specifications are either declarative or procedural.

  — A declarative mapping specification identifies a pattern of instances of one model that corresponds to a pattern of instances in another model. If the mapping is correct, the mapping is bi-directional.

  — A procedural mapping defines the process by which an instance of the target model can be created from instance(s) of the source model. For procedural mappings, a mapping in each direction is required.

6 Overview of the model integration process

The model integration process takes a number of application models and an integration model. It ensures that all the concepts of the application models are represented in the integration model, and develops a mapping specification between the integration model and each of the application models.

There are three possible cases for the integration process:

— the integration model and the application models both exist before the integration process starts;

— the application models to be integrated exist before the integration process starts, but not the integration model;

— the integration model exists before the integration process starts, but the application model needs to be developed from some statement of requirements.

The first of these covers all the elements of the other two, and is described here in outline for one application model. The other two processes are described in more detail in ISO18876-2.

Integrating an application model with an integration model is illustrated in Figure 9 below. The goal of this integration process is to allow the same information that is represented in the application models to be represented in the integration model without loss of meaning, and to allow transformations between these representations. The result of the integration process is a mapping specification between the application model and a part of the integration model. In order to define this mapping it may be necessary to extend the integration model so that it precisely represents the concepts found in the application model.

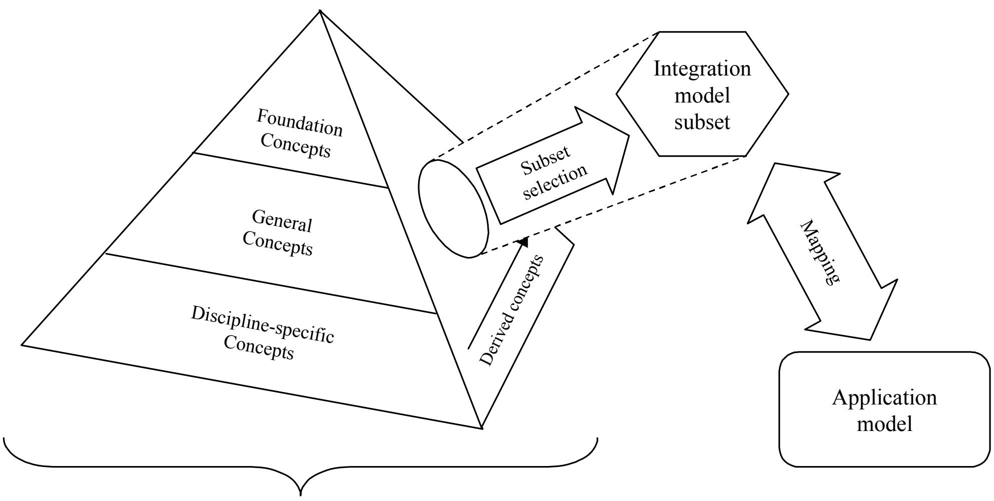

Figure 9 — Integrating application models with an integration model

Being an integration model derives from the role that it plays with respect to other application models. The fundamental characteristic of an integration model is that it integrates two or more application models.

The process of integrating an application model with an integration model is divided into a number of steps, as follows:

— analyse the application models and identify the equivalent concept of the integration model, including any constraints that apply, see Figure 10;

NOTE Most application models have a context within which the model has to be understood, but which is not explicit in the model itself. Usually it will be inappropriate to add this information explicitly to the application model. In this case these requirements should be captured in the mapping specification as part of the integration process.

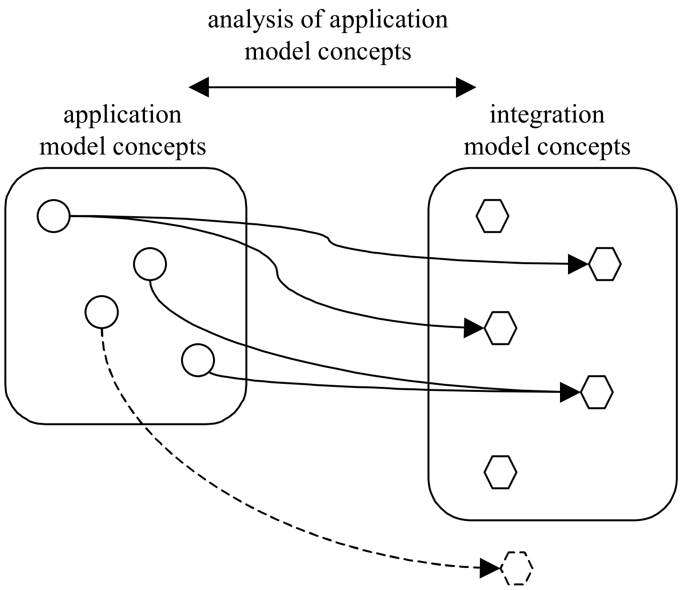

Figure 10 — Analyzing the application models

— if necessary, extend the integration model so that it includes all the concepts found in the application models, see Figure 11;

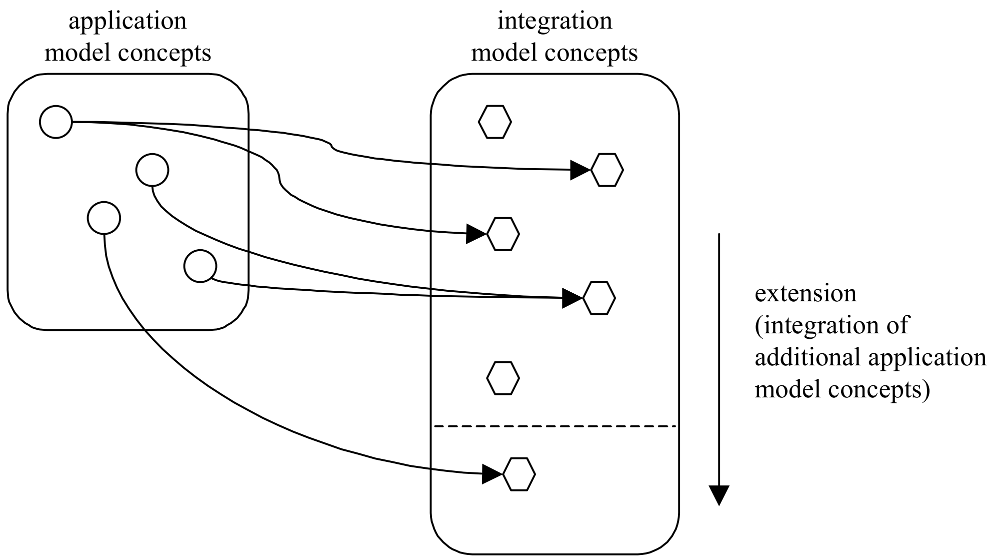

Figure 11 — Adding any missing concepts to the integration model

— identify the part of the integration model that represents the concepts in each application model, see Figure 12;

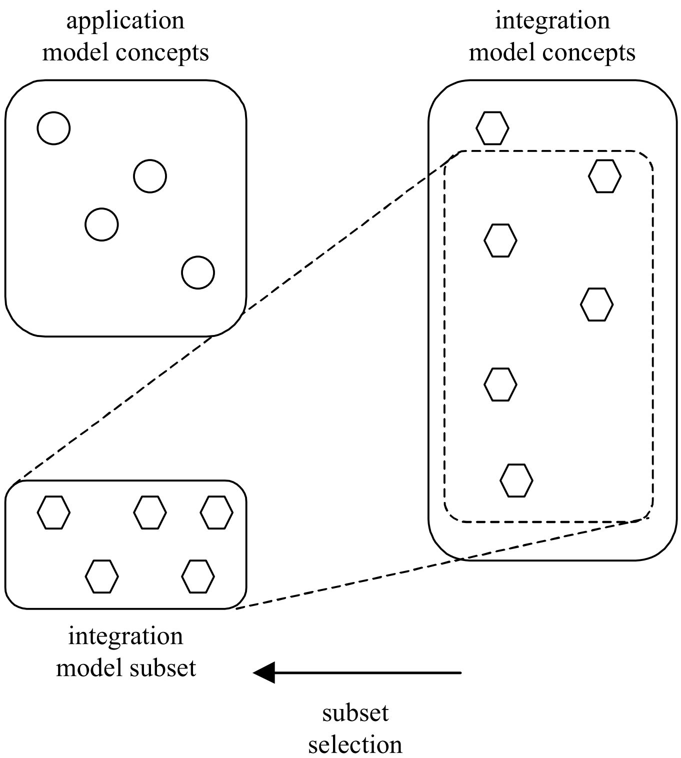

Figure 12 — Identifying the subset of the integration model

— create the mapping in each direction between each application model and the appropriate subset of the integration model, see Figure 13;

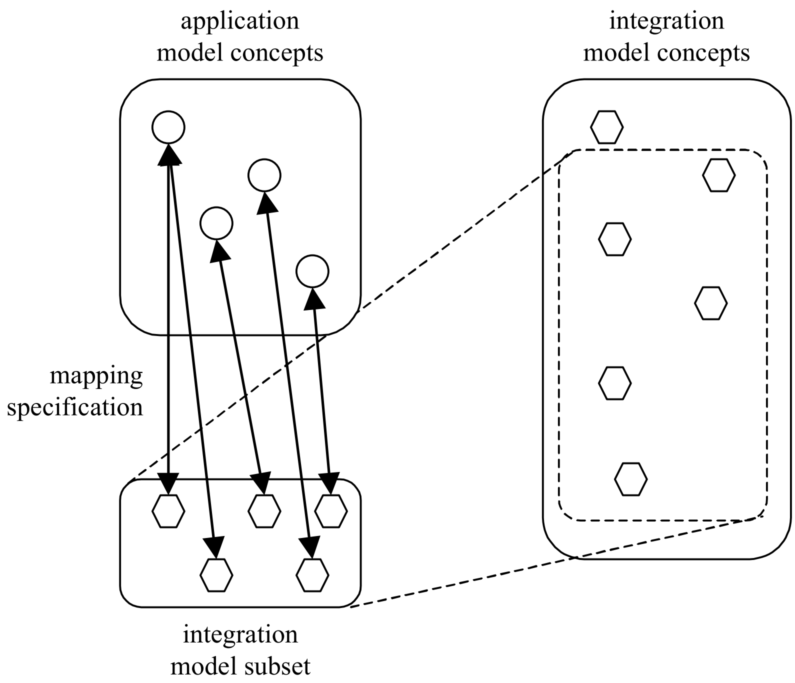

Figure 13 — Creating the mapping between the integration model subset and the application model

— specify any structural transformations, terminology transformations, or encoding transformations that apply within the mapping;

— specify any transformations that are necessary between model representations;

EXAMPLE 1 If an application model is specified in the XML Schema definition language and the integration model to which it is mapped is specifed in EXPRESS (ISO 10303-11), a transformation between these languages will be necessary to map between different representations of the same concepts.

— repeat this process for all other application models to be integrated.

Most application models have a context within which the model has to be understood, but which is not explicitly represented in the application model itself. Mapping successfully in both directions requires that both the explicit model and its context be mapped into the integration model.

EXAMPLE 2 In a salary payment system may be an entity data type called **employee.** However, it is often implicit that each person represented by instances of this entity data type is an employee of the company that operates the system and legally eligible for employment under company and governmental policies.

7 Integration architecture components

Figure 14 gives an alternative view of Figure 9 for entity attribute relationship modelling languages and models.

NOTE 1 In this International Standard object oriented models are considered as entity attribute relationship models.

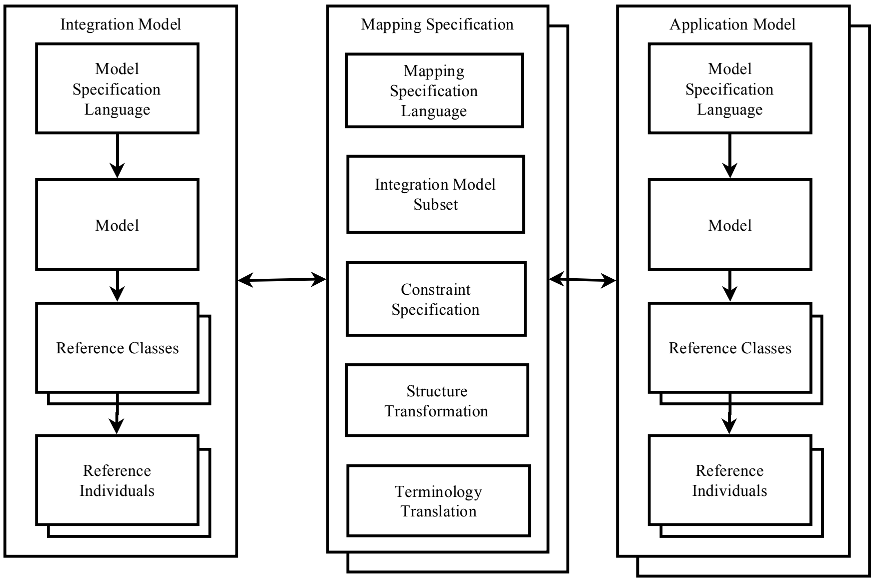

Figure 14 — Integration architecture components

Figure 14 shows the elements that may be standardized, as follows:

— integration models;

NOTE 2 This may be a single model with multiple levels of abstraction, using a suitable logic based language, such as KIF or EXIST, or a layered model, using an entity-relationship language such as EXPRESS with a data model and reference data libraries. Figure 14 illustrates the use of a model to define the structure of a reference data library that can hold reference classes and reference individuals.

NOTE 3 There may be a number of reference data libraries. These are to cover the discipline specific primitive concepts, and derived concepts. Procedures may be required for their development to ensure there is no duplication across the libraries.

— mapping specifications that specify a part of the integration model, constraints that specify the subset of the integration model that maps to the application model, structural transformations between the structure of the integration model and the structure of the application model, and finally terminology transformations between the integration model and an application model;

NOTE 4 Mapping specifications may also be required between model specification languages.

— application models.

8 Data mapping and consolidation

Mapping between models is not sufficient to achieve integration. This requires reconciliation of information represented according to the different models. This process is illustrated in Figure 15 below.

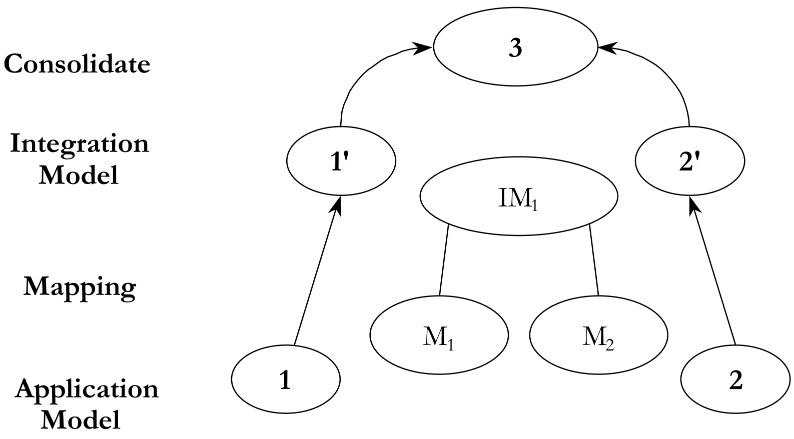

Figure 15 — Data consolidation

— Translate the data population 1, and 2, according to their source models into the data populations 1' and 2' according to the model IM_1.

— Identify which data elements in the two data sets represent the same things, and consolidate them in data population 3.

NOTE This requires a common, reliable, persistent identification mechanism. This does not necessarily mean defining some new attribute for use by the various systems. It may be possible to use an existing attribute. There may be a collection of attributes and relationships that together can provide a unique identification. Any of these may be mapped to an independent identification scheme.

9 Relationship to other standards

This International Standard can be used in conjunction with other standards. It can be used by a standard that wishes to integrate a set of models to create that integration by design.

EXAMPLE 1 ISO 15926 [5] specifies a data model and a reference data library that together form an integration model for process industry applications.

This International Standard can also be used to integrate existing information standards where integration is desired but was not achieved by design.

EXAMPLE 2 The methodology described in ISO 18876-2 can be used to integrate a product data exchange capability specified in an ISO 10303 application protocol and a document exchange capability described by an XML Document Type Definition.

Annex A
(normative)

Information object registration

To provide for unambiguous identification of an information object in an open system, the object identifier

`{iso standard 18876 part{1} version {1}}`

is assigned to this part of ISO 19976. The meaning of this value is defined in ISO/IEC 8824-1, and is described in ISO 10303-1.

NOTE This is the object identifier that will apply to the published (IS) version of this part of ISO 18876.

The reference to Part 1 of STEP assumes that the second edition of Part 1 defines usage of ASN.1 identifiers that is not limited to ISO 10303.

Bibliography

[1] ISO 8879:1986, *Information processing – Text and office systems – Standard Generalized MarkupLanguage.*

[2] ISO/TR 9007:1988, *Information processing systems -- concepts and terminology for the conceptual schema and the information base*.

[3] ISO 10303-11:1994, *Industrial automation systems and integration — Product data representation and exchange — Part 11: The EXPRESS language reference manual.*

[4] ISO 10303-21:1994, *Industrial automation systems and integration — Product data representation and exchange — Part 21: Implementation methods: Clear text encoding of the exchange structure.*

[5] ISO15926-2:—[2), *Industrial automation systems and integration — Integration of life-cycle data for process plants including oil and gas production facilities — Part 2: Data model*.

[6] ISO/IEC 19501-1: —[3), *Information technology — Unified Modeling Language (UML) — Part 1: Specification.*

[7] *Extensible Markup Language (XML) 1.0 (Second Edition)*. W3C Recommendation 6 October 2000 [cited 2001-05-16]. Available from the Internet: <http://www.w3.org/TR/REC-xml>

[8] *Integration Definition for Information Modeling (IDEF1X).* Federal Information Processing Standards Publication 184, December 1993

[9] BARWISE, Jon; SELIGMAN, Jerry. *Information flow: the logic of distributed systems*. Cambridge: Cambridge University Press, 1997.

[10] FOWLER, Julian. *Industry requirements for SC4 standards*. ISO TC184/SC4/WG10 N173, 1998 [cited 2001-05-09]. Available from the Internet: <http://www.nist.gov/sc4/wg_qc/wg10/current/n173/wg10n173.htm>.

[11] GUARINO, Nicola. *Some Organizing Principles for a Unified Top Level Ontology.* Padua: 1997 [cited 2001-05-09]. Available from the Internet: <http://www.ladseb.pd.cnr.it/infor/Ontology/Papers/TopLevel.pdf>.

[12] GUARINO, Nicola. *Some Ontological Principles for Designing Upper Level Lexical Resources*. Padua: 1998 [cited 2001-05-09], available from the Internet: <http://www.ladseb.pd.cnr.it/infor/Ontology/Papers/LREC98.pdf>

[13] PARTRIDGE, Chris. *Business objects: re-engineering for reuse*. Oxford: Butterworth Heinemann, 1996.

[14] SOWA, John F. *Knowledge representation: logical, philosophical, and computational foundations*. Pacific Grove CA: Brooks/Cole, 2000.

[15] WEST, Matthew; FOWLER, Julian. *Developing High Quality Data Models*. Version 2.0, Issue 2.1. EPISTLE, 1996 [cited 2001-05-09]. Available from the Internet: <http://www.stepcom.ncl.ac.uk/epistle/data/mdlgdocs.htm>.

---

2) To be published

3) To be published

Index